Tolerating Adversarial Reviews

Nithya Balaji | Joshua Turner
Clarence Lam | Emily Inkrott

Mentors: Jonathan Katz, Benjamin Sela

REU-CAAR 2021

DEPARTMENT OF
COMPUTER SCIENCE

Motivation

Facebook - Real News vs. Fake News?

- Wished to stop the spread of fake news
- Had difficulty correctly identifying fake news

Our Problem

Let there be a set of *m* reviewers reviewing *n* items. A reviewer can either review an item as “bad” (0) or “good” (1).

How can we estimate the true quality (good or bad) of the items based on the set of reviews?

- One approach: take the majority of all of the reviews for each item

What if some of the reviews are generated by an adversary who wants to maximize how many items we incorrectly estimate?

Our Problem

Two types of reviewers: honest and malicious

- m: number of reviewers
- n: number of items being reviewed
- α: fraction of the reviewers that are maliciously generated ($\alpha < 0.5$)
- p: probability of a honest review being correct on any given item ($p > 0.5$)
- q: probability of a honest review being incorrect on any given item ($1-p$)

How can we correctly estimate the true quality of an item without being biased by adversarial reviews?

- By identifying and eliminating adversarial reviews, then taking the majority of the remaining reviews for each item.

Overview

Robust Estimation Algorithm

Active Learning Approaches

The Algorithm

- The "Naive" Algorithm
- Benjamin Sela's Algorithm
- Performance
- Improving the Algorithm

The "Naive" Algorithm

Algorithm

- For each item, return a majority vote over all reviews

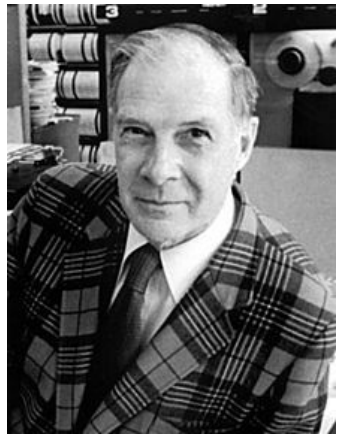
Analysis:

- For a given item, a p-fraction of the $(1-\alpha)m$ honest reviews will be correct
- We want at least $m/2$ reviews to be correct, which constrains $p > 1/(2(1-\alpha))$
- Worst case: Adversarial reviews are incorrect on every item

Sela's Algorithm

- We expect all honest reviews to have a Hamming distance of approximately $2pqn$ from each other
- Eliminate reviews that are farther away than expected
  - Reviews with a distance of over $(1+\varepsilon)2pqn$ from $(1-\alpha)m$ reviews are eliminated
  - An adversarial review cannot be too far away from the correct values or it will be eliminated
- Eliminate reviews that are closer together than expected
  - Pairs of reviews having a distance of less than $(1-\varepsilon)2pqn$ are eliminated
  - Adversarial reviews cannot be too close to each other or to an honest review
- Take a majority vote over the remaining reviews

Richard Hamming



Performance

How does Sela's algorithm perform compared to the naive algorithm?

- Test both algorithms against different adversarial strategies
  - Compare accuracies at varying p values and α values for smaller and larger number of reviews and items.
  - Check the percentage of honest and adversarial reviews that are removed by Sela's algorithm.

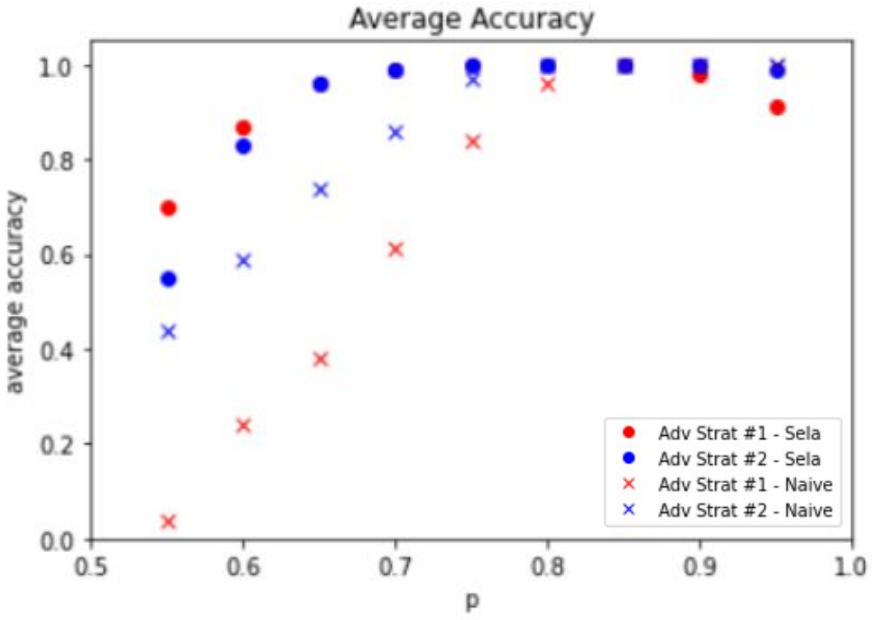
Two adversarial strategies:

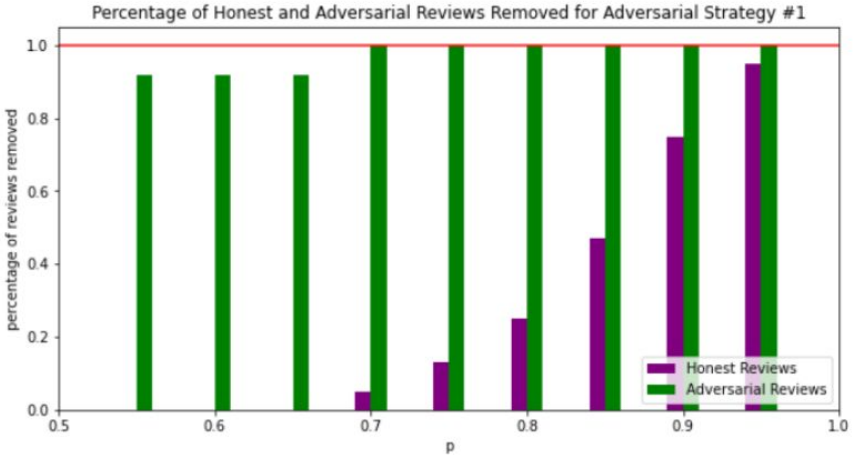
- Adversarial Strategy #1
  - The adversary is incorrect on every item.
- Adversarial Strategy #2
  - The adversary is incorrect on some items. Enough of the adversarial reviewers are made to be incorrect on a subset of items so that the majority of all of the reviews for that item would be incorrect. The remaining reviewers would act honestly for that item.

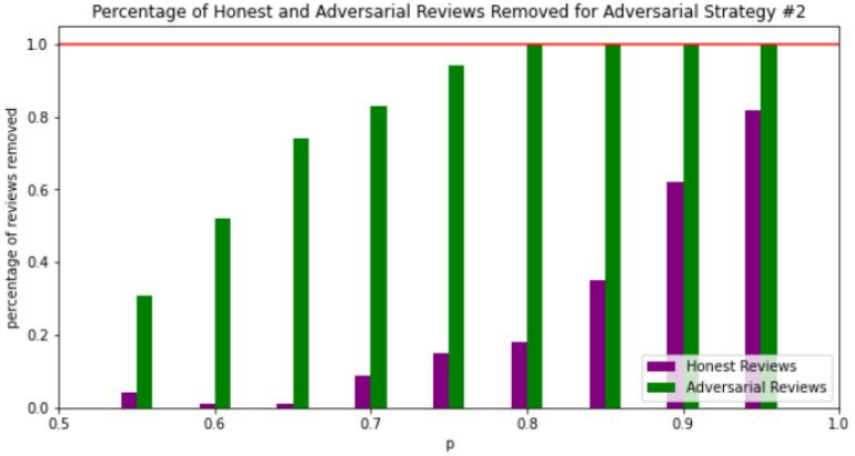
Performance

- Based on the experimental results, Sela's algorithm performs better for a larger number of reviews.
- Sela's algorithm performs better than the naive algorithm when the probability of the honest reviewers being correct (p) is smaller and when the fraction of adversarial reviewers (α) is larger.
  - The threshold for p is approximately $p > 0.8$
  - The threshold for α is approximately $\alpha < 0.25$

Varying *p* - smaller number of reviewers

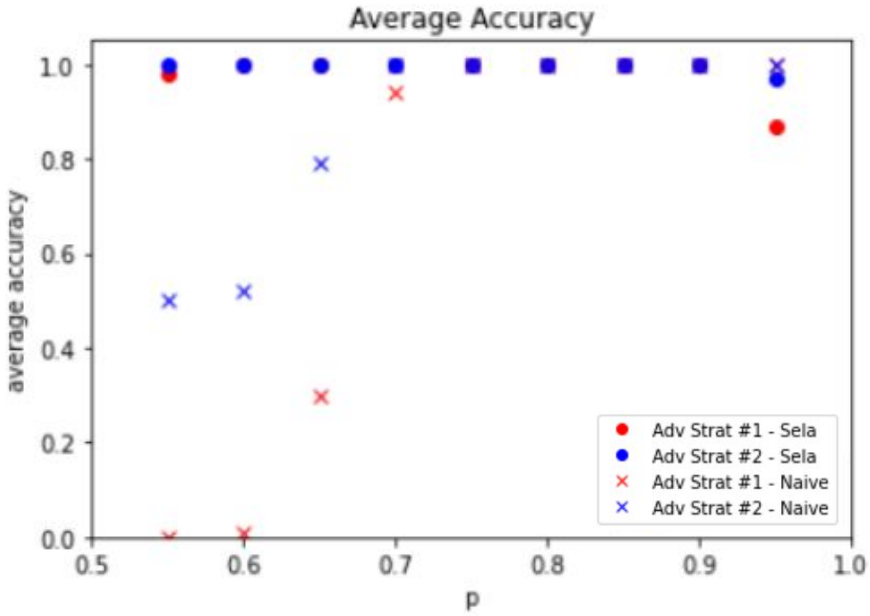


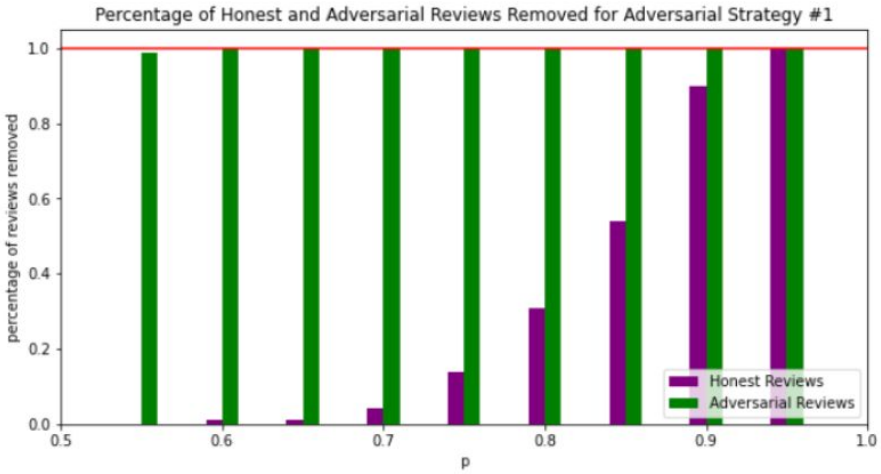


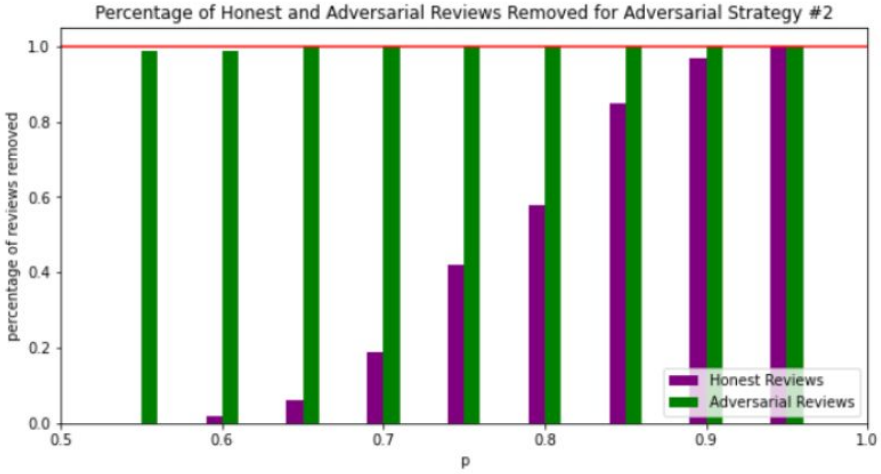


α = 0.25, *m* = 50, *n* = 50, ε = 0.5

Varying *p* - larger number of reviewers

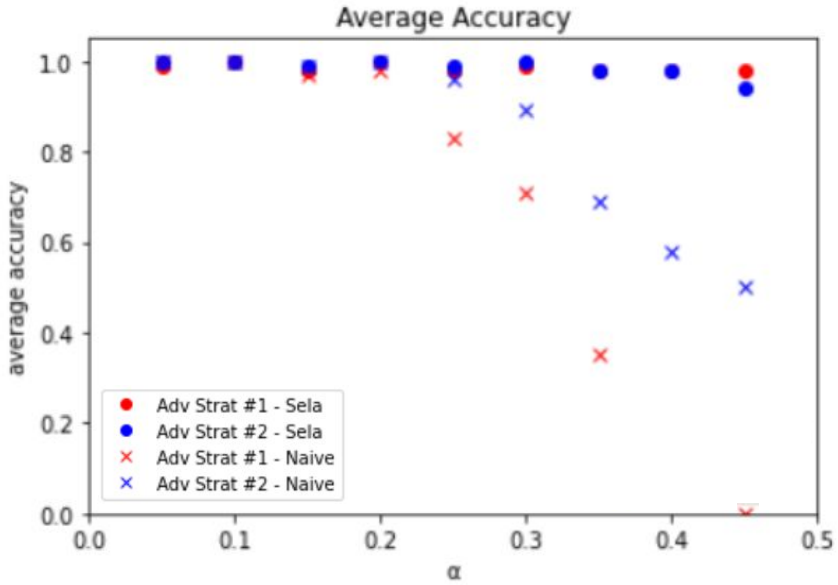


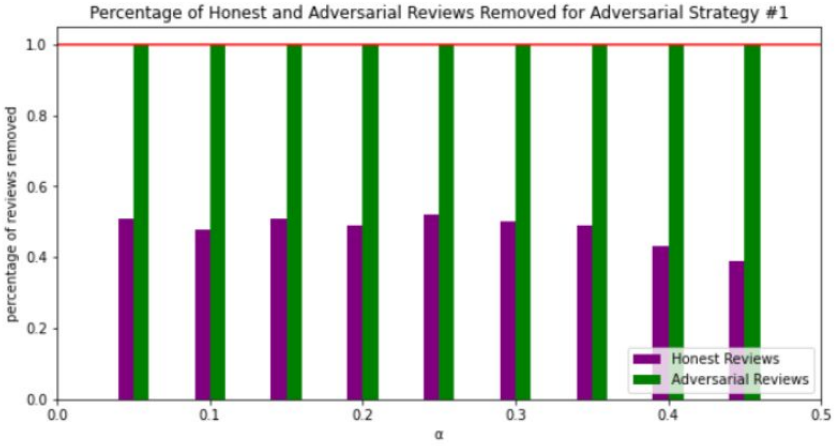


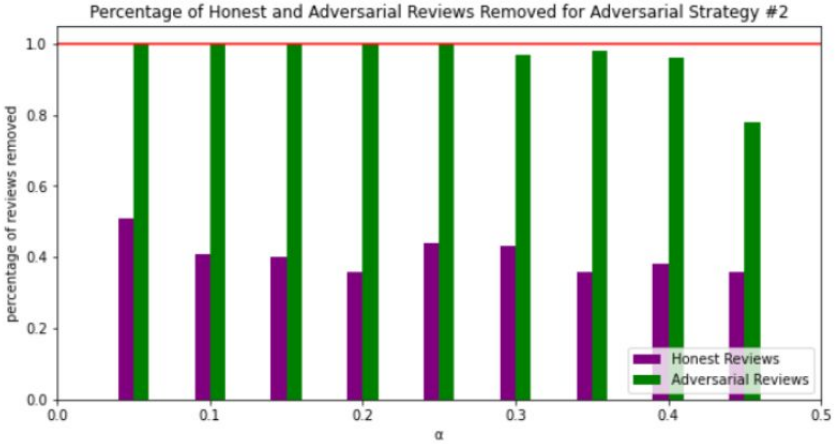


α = 0.25, *m* = 500, *n* = 500, ε = 0.2

Varying α - smaller number of reviewers

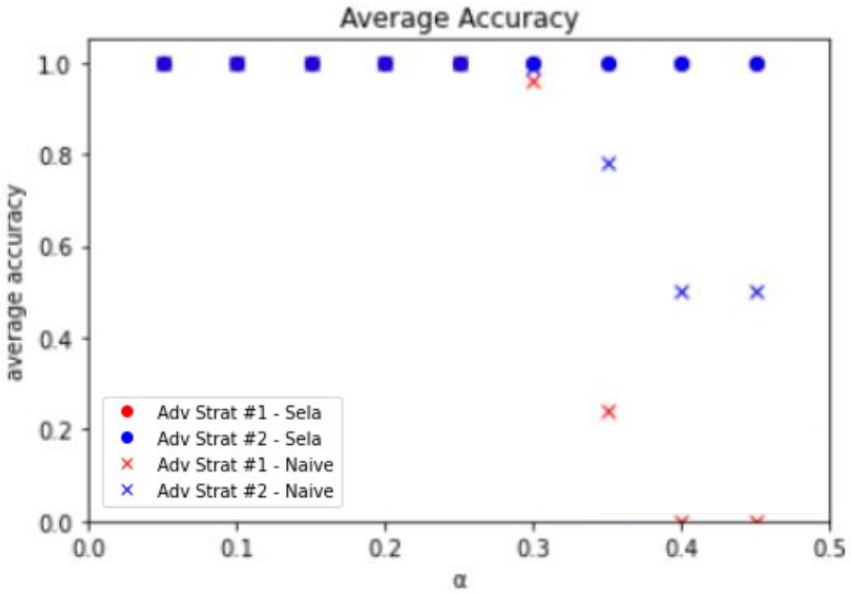


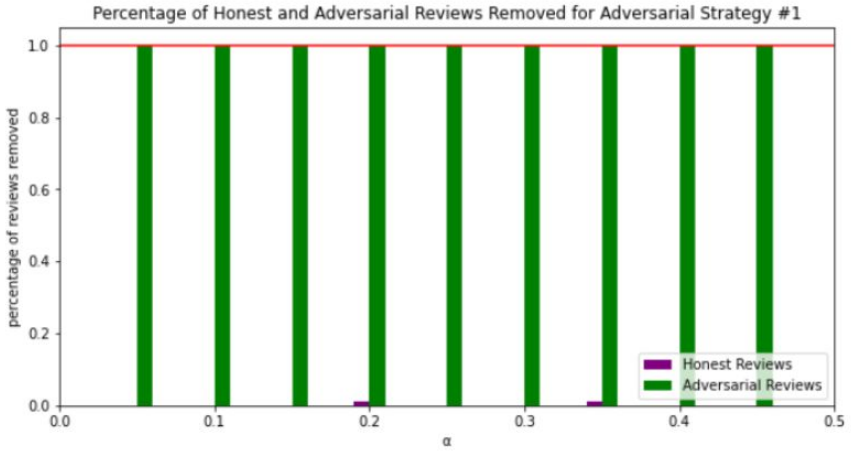


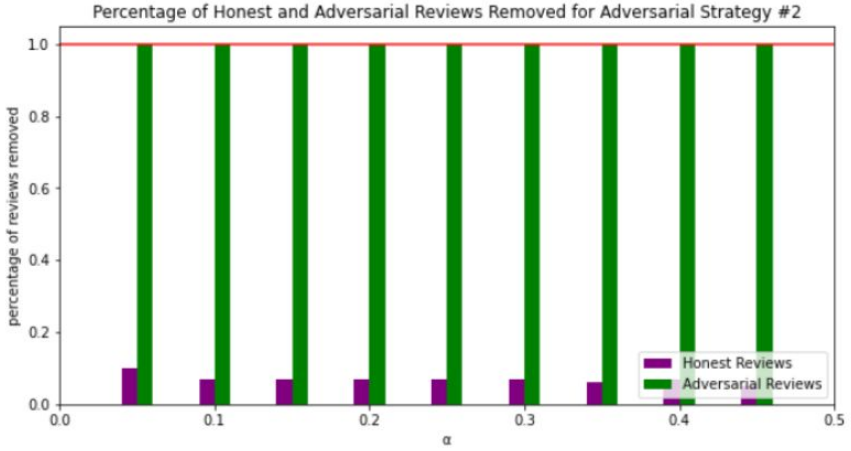


p = 0.75, m = 50, n = 50, ε = 0.4

Varying α - larger number of reviewers







p = 0.75, m = 500, n = 500, ε = 0.25

Improving the Algorithm

- Problem: The algorithm can perform worse at very large values of m
- Solution 1: Let ε depend on m
  - This may not work for very large m since at that point, two honest reviews will be exactly the same
- Solution 2: don't work with large m, work only in groups of size k
  - If we partition the m reviews into groups, we are guaranteed that a majority of groups have a majority of honest reviews
- Modified algorithm
  - Split the reviews into groups of size k
  - Run the current algorithm on each group
  - Take a majority from the results

Active Learning

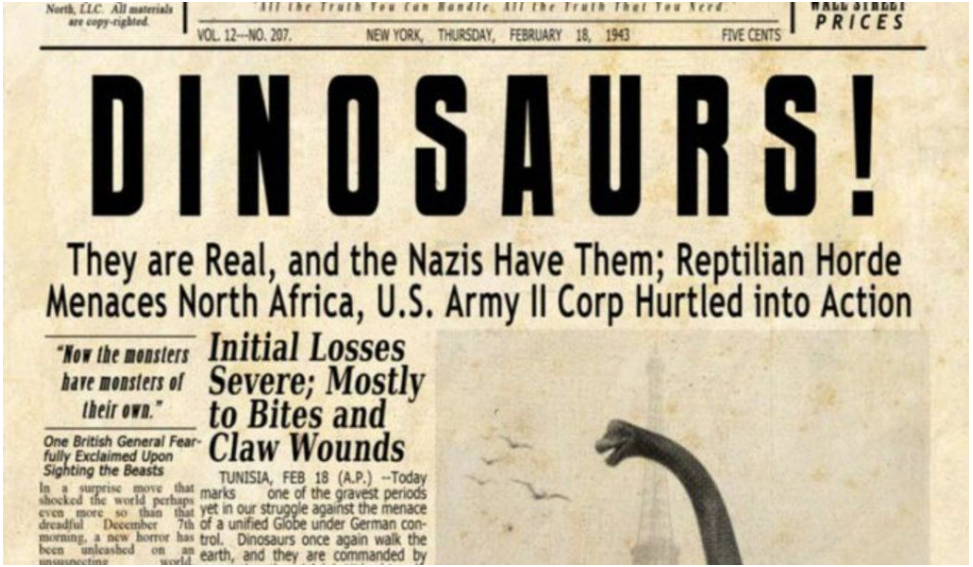
What if we can query the true values of some small number of items?

Key Questions

- *Which items should we learn about?*
- *What should we do with this information?*
- *How much can it help us?*

Motivation (Reprise)

- Fake news: fact-checking agencies
- Accuracy on checked items and overall accuracy



North, LLC. All materials are copy-righted.

VOL. 12—NO. 207. NEW YORK, THURSDAY, FEBRUARY 18, 1943 FIVE CENTS

PRICES

DINOSAURS!

They are Real, and the Nazis Have Them; Reptilian Horde Menaces North Africa, U.S. Army II Corp Hurtled into Action

"Now the monsters have monsters of their own."

One British General Fearfully Exclaimed Upon Sighting the Beasts

In a surprise move that shocked the world perhaps even more so than that dreadful December 7th morning, a new horror has been unleashed on an

Initial Losses Severe; Mostly to Bites and Claw Wounds

TUNISIA, FEB 18 (A.P.) --Today marks one of the gravest periods yet in our struggle against the menace of a unified Globe under German control. Dinosaurs once again walk the earth, and they are commanded by

Example Rules

Random item

0	1	0	0	1	0	1	1	1
72%	67%	74%	68%	52%	70%	61%	70%	66%

Closest to 50/50

0	1	0	0	1	0	1	1	1
72%	67%	74%	68%	52%	70%	61%	70%	66%

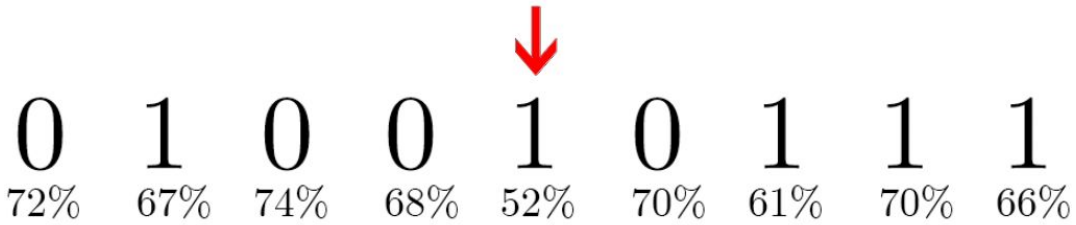
Why 50/50?

- Item we're least certain about
- Most guaranteed wrong answers
- Honest sufficiently correct -> discover malicious reviews

Deterministic 50/50 Query

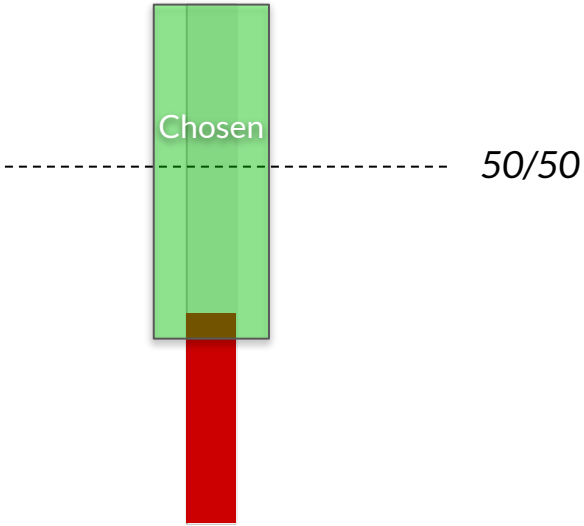
Algorithm

1. Query the item which is closest to a 50/50 split
2. Throw out all reviewers which are incorrect on that item
3. Take a majority vote over the remaining reviews



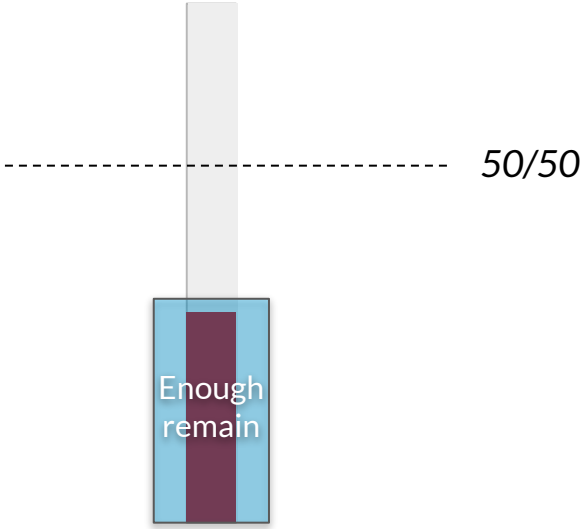
Attacking 50/50

Item needs to be *close* enough to 50/50 to get chosen...



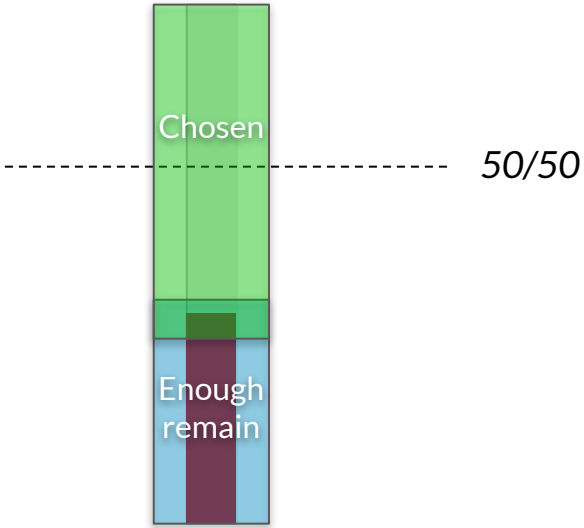
Attacking 50/50

…but *far* enough from 50/50 so that adversary doesn't lose too many reviews.



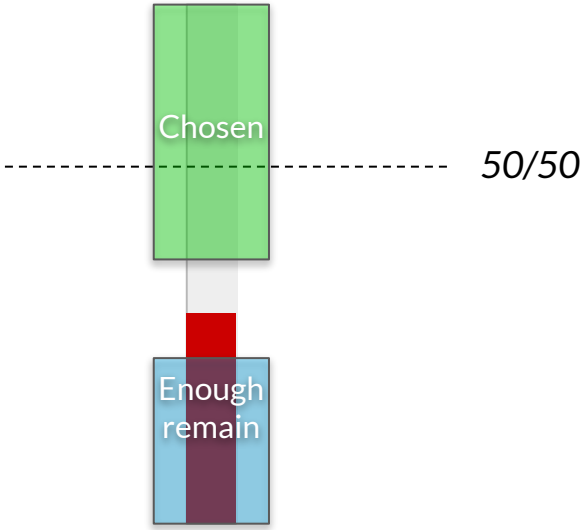
Attacking 50/50

For attack success, # reviewers wrong on queried item must be in the *intersection*.



Attacking 50/50

Disjoint means attack is *impossible*.



Attacking 50/50: The Inequalities

$$\alpha > 1 - \frac{1}{2p^2 + p}$$

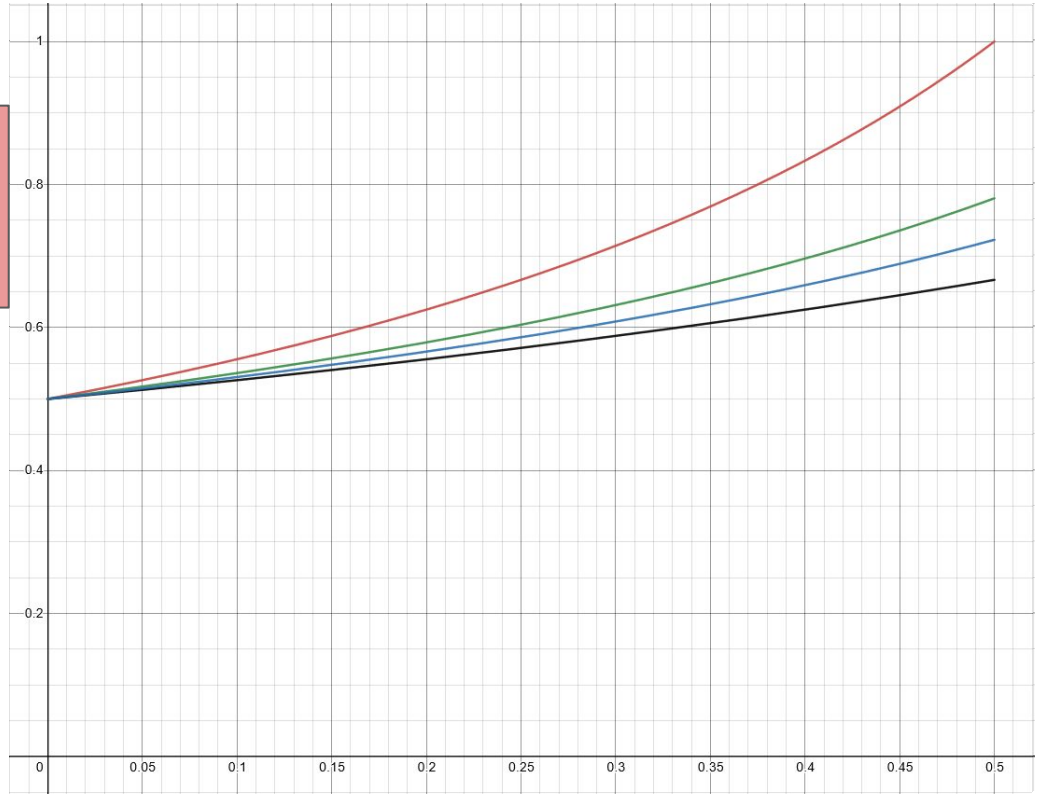
$$\alpha > 1 - \frac{q}{p\left(1 + p^K - 2p^{K+1}\right)}$$

Notation Key

α - % reviews adversary controls
p - Pr(honest reviewer correct on item)
q - 1-*p*
K - # of 50/50 queries

Impossibility Results: When is attack impossible?

Notation Key
α - % reviews adversary controls
p - Pr(honest reviewer correct on item)
q - 1-*p*
K - # of 50/50 queries



K = 0 (*p*≥1)

K = 1 (*p*≥0.78)
K = 2 (*p*≥0.72)
K = ∞ (*p*≥0.66)

How accurate honest reviewers must be to prevent attack ($p > ½$)

Proportion of reviews that are adversarial ($\alpha < ½$)

What about items that aren't at 50/50?

If the adversary is dedicated, they can flip an item by more than a 50/50 split. This is easier the more adversaries there are.

Probabilistic queries help us to check items that aren't necessarily at 50/50, with a higher probability for items with a more highly contested vote.

This gives us a chance to catch adversaries on items that were not successfully flipped, or were flipped by a large margin.

Probabilistic 50/50: Which items to query?

We can determine the probability ϕ that a given item will be queried with a simple equation.

$$\phi = 2(\text{minority})$$

The closer an item is to 50/50, the more likely it is to be queried.

Alternate to Elimination

Rather than eliminate any incorrect reviewer, we can lower their impact on the vote.

We accomplish this by **halving the weight** of a reviewer's vote when they are incorrect on a queried item.

Pro: less damage to honest reviewers

Con: "getting caught" doesn't completely stop an adversary.

Combining this with other adjustments has interesting results.

Conclusion

This summer, we…

- Considered improvements for a robust estimation algorithm
- Measured empirical performance of the original algorithm
- Investigated active learning approaches

Thank you!